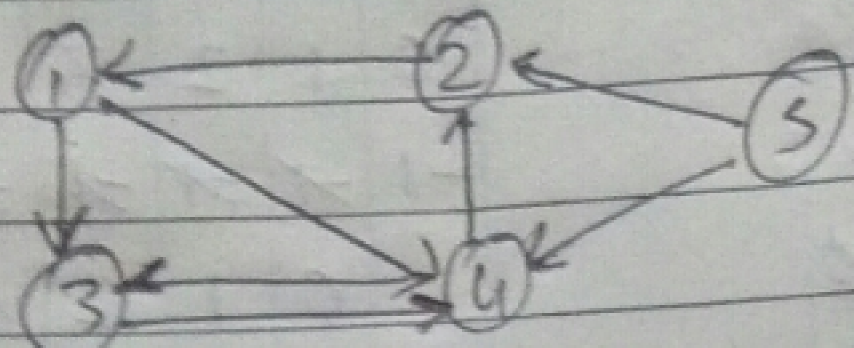
⑥ Floyd Warshall algo.



$$\text{dist} \longrightarrow \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & \infty & 6 & 3 & \infty \\ 2 & 3 & 0 & \infty & \infty & \infty \\ 3 & \infty & \infty & 0 & 2 & \infty \\ 4 & 0 & 1 & 1 & 0 & \infty \\ 5 & \infty & 4 & \infty & 2 & 0 \end{array}$$

$$\text{list } 1 = \begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & \infty & \infty & \infty \\ \infty & \infty & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix} \qquad \text{list } 2 = \begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & 7 & 6 & \infty \\ \infty & \infty & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix}$$

$$\text{list } 3 \rightarrow \begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & 7 & 6 & \infty \\ 2 & 3 & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix} \qquad \text{list } 4 = \begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & 2 & 6 & \infty \\ 2 & 3 & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ \infty & 4 & \infty & 2 & 0 \end{bmatrix}$$

$$\text{list } 5 \rightarrow \begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & 7 & 6 & \infty \\ 2 & 3 & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ 2 & 3 & 3 & 2 & 0 \end{bmatrix}$$

final matrix ~~[illegible]~~ shortest dist b/w every pair

$$\begin{bmatrix} 0 & 4 & 4 & 3 & \infty \\ 3 & 0 & 7 & 6 & \infty \\ 2 & 3 & 0 & 2 & \infty \\ 0 & 1 & 1 & 0 & \infty \\ 2 & 3 & 3 & 2 & 0 \end{bmatrix} \quad \text{Ans}$$

$$\boxed{TC = O(V^3)}$$

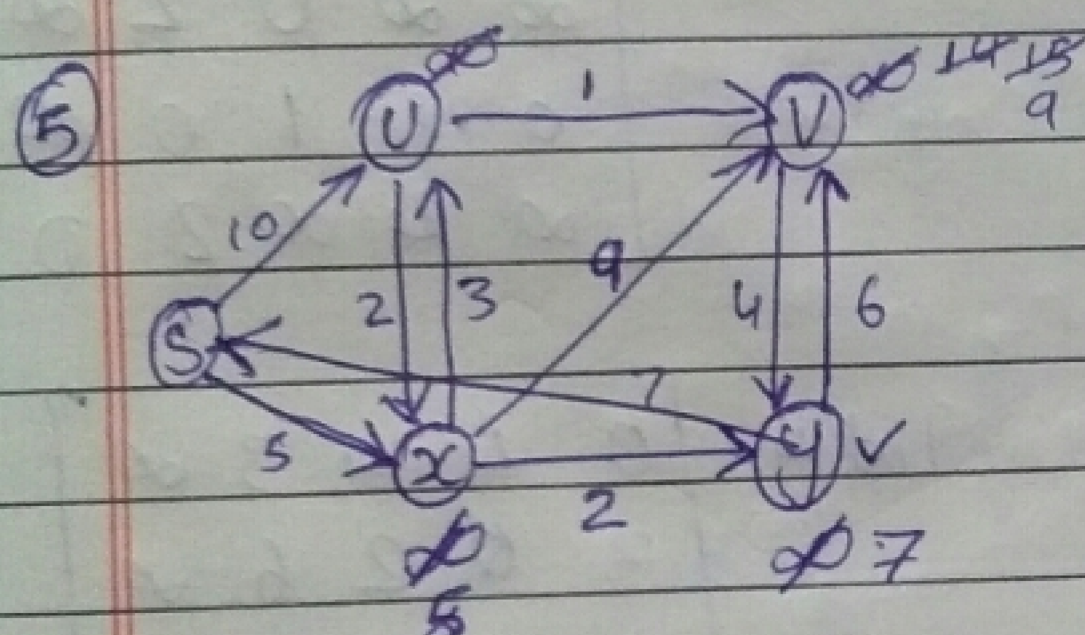
Weight table :

0	1	2	3	4	5	6	7	8
~~∞~~	~~∞~~	~~∞~~	∞	∞	∞	∞	~~∞~~	∞
0	4	8	~~∞~~	∞	~~∞~~	∞	8	~~∞~~
			7	∞	4	~~∞~~	~~8~~	2
			7	~~∞~~	4	~~6~~	7	
			7	10		~~2~~	7	
			7	~~10~~		1		
				9				

Rut ch 7

0	1	2	3	4	5	6	7	8
-1	~~-1~~	~~-1~~	~~-1~~	~~-1~~	~~-1~~	~~-1~~	-1	~~-1~~
	0	1	2	~~5~~	2	~~8~~	~~0~~	2
				3		~~5~~	8	
						7		

(5)



Dijkshis Algo →

node	short distance
U	8
V	9
x	5
y	7

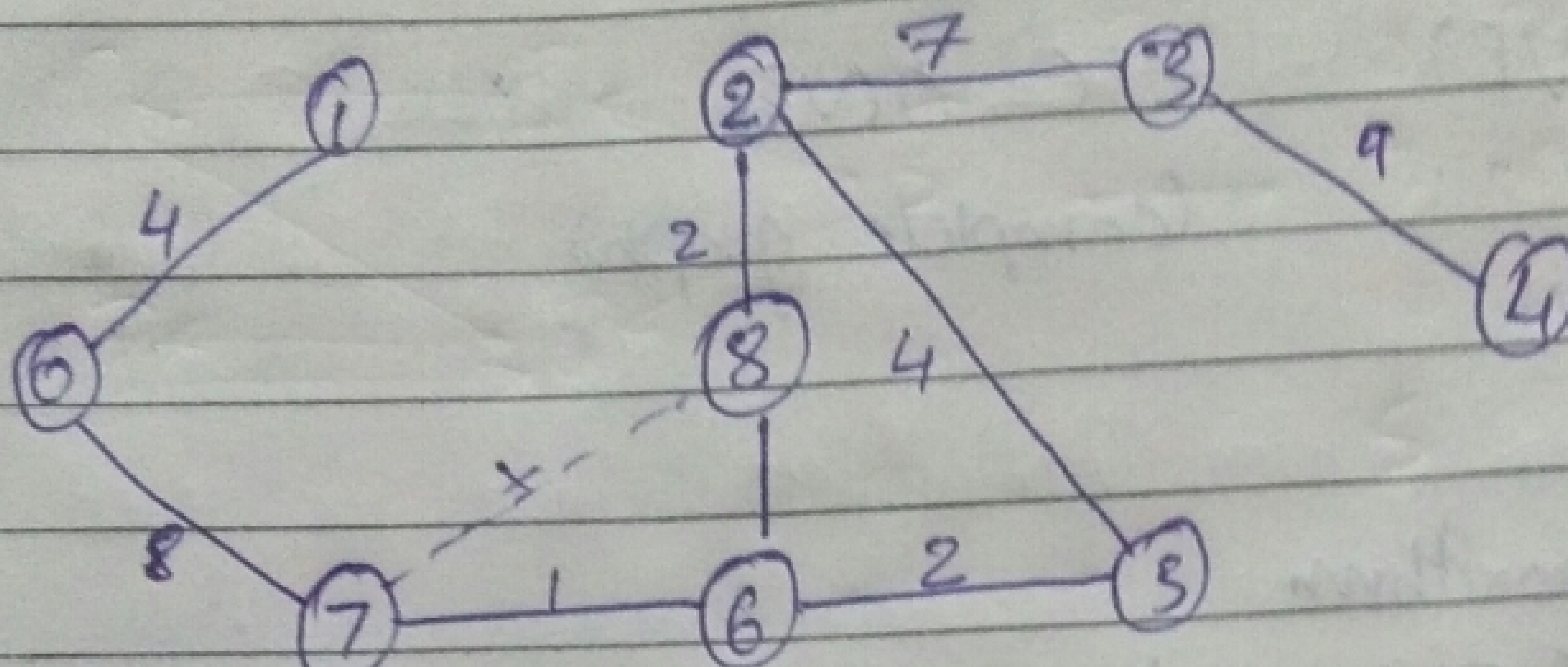
Bellmanford

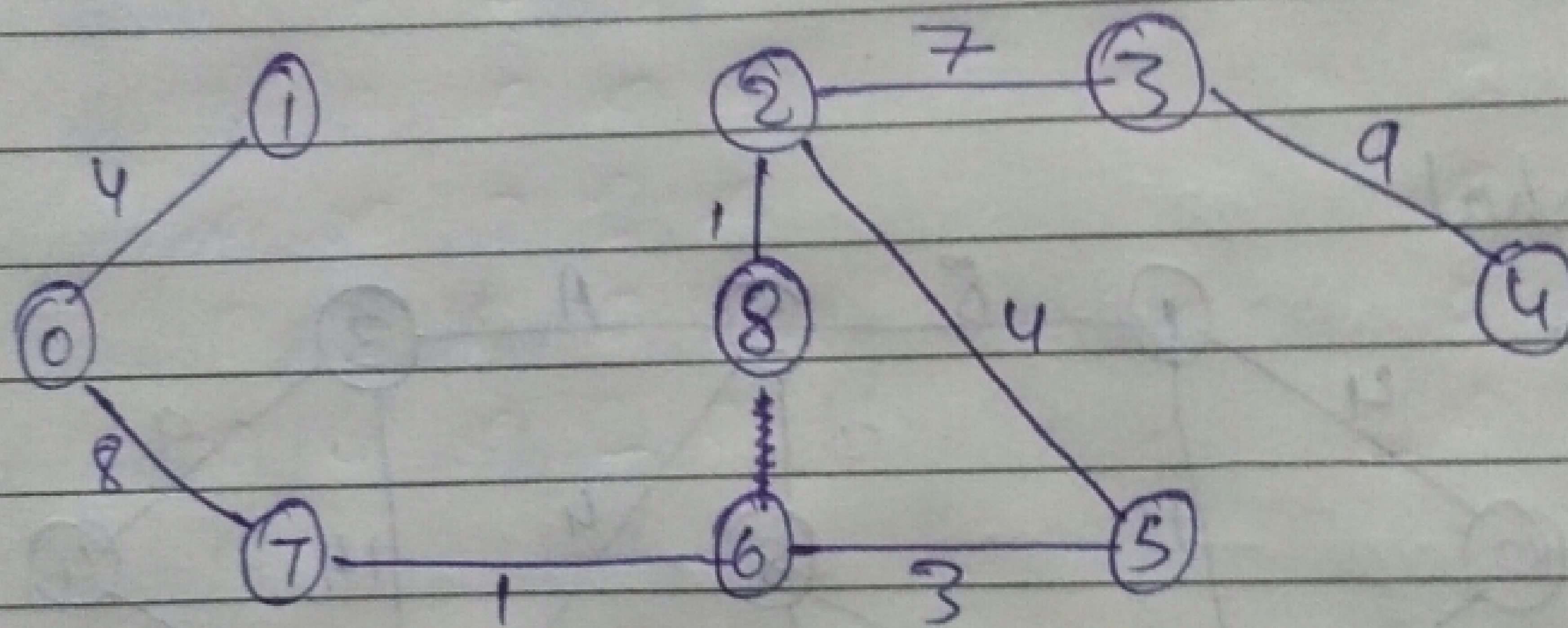
Node / Iter	S	U	V	X	Y
Iteration 1	0	~~∞~~ ~~9~~ 8	~~∞~~ 9	~~∞~~ 5	~~∞~~ 7
Iteration 2	0	8	9	5	7
Iteration 3					
Iteration 4					

After 2nd interation we that then has been no change in

node	Shortest path
U	8
V	9
X	5
Y	7

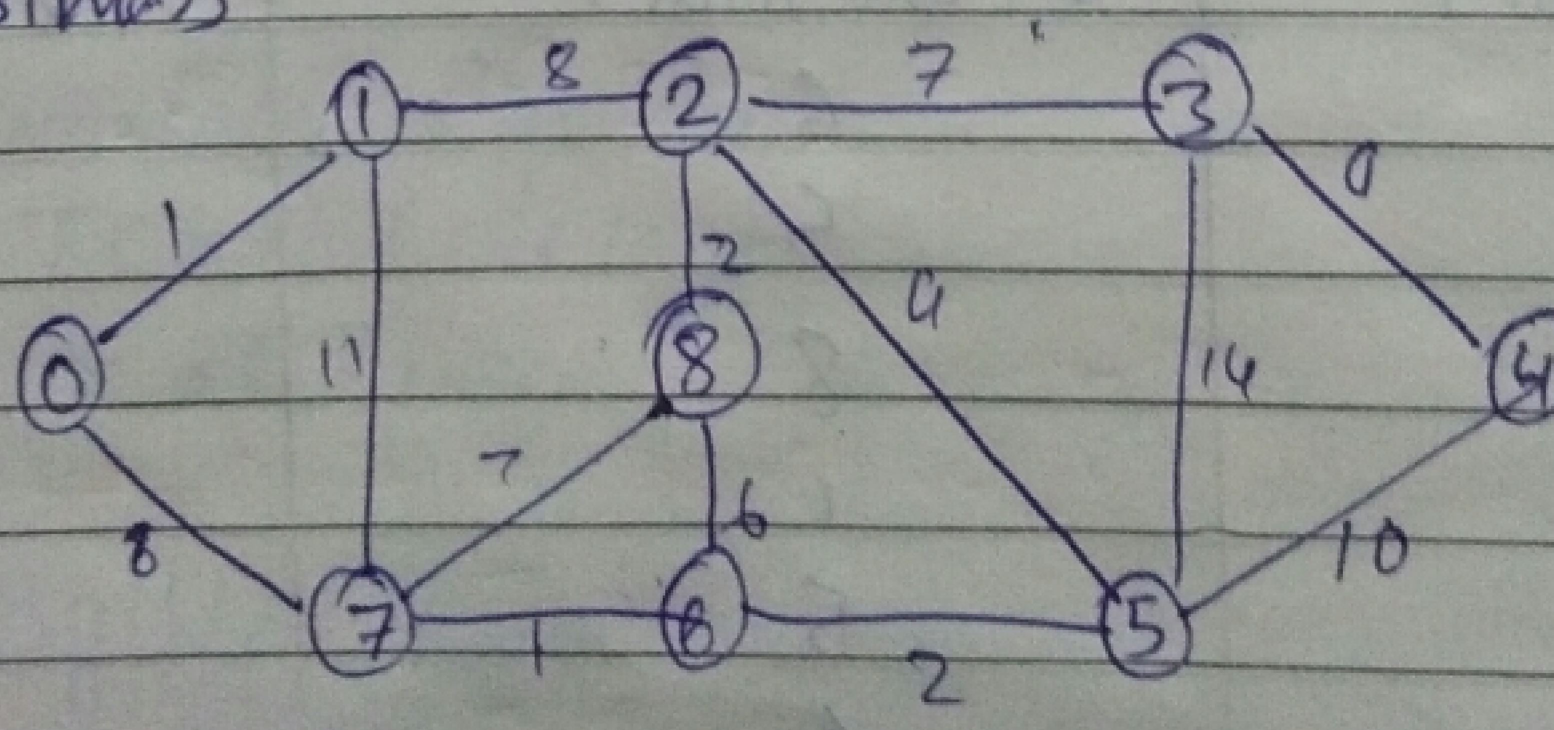
Put the edge





∴ Total weight = 37

(2) Prim's



③ Bellman Ford's Algorithms

TC = O(VE)   TC = O(V^3)
(complete graph)

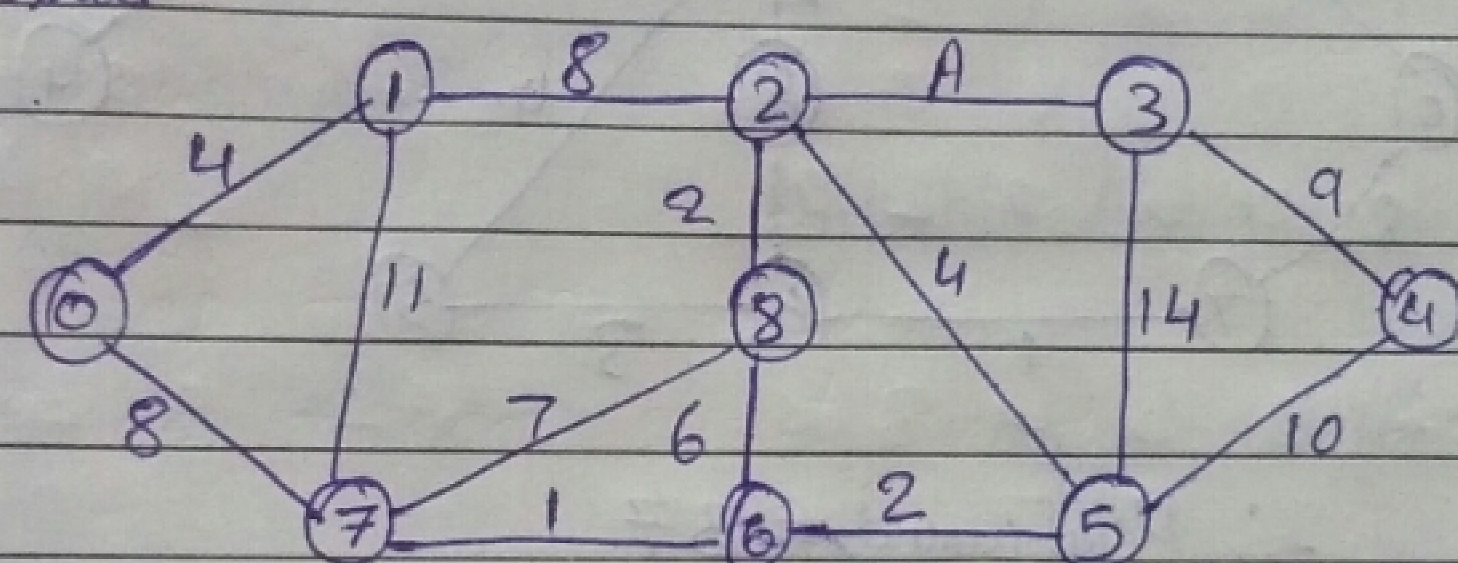
SC = O(V)

④ Mink's Algorithm

TC = O(|E| log |V|)

SC = O(|V|)

Q 3 Krushal



U (1st Node)	V (2nd node)	U-V edge weight
7	6	1
6	5	2
2	8	2
0	1	4
2	5	4
8	6	6
8	3	7
7	8	7
0	7	8
1	2	8
3	4	9
5	4	10
1	7	11
[illegible]	5	14

Date
Page

DAA Tutorial 6

NAME - TARUN KUMAR DHIMAN
ROLL NO - 1918917
Sec - IT

① A minimum spanning tree is a tree is which the sum of the weight of the edges is as minimum as possible, connecting all the vertices a the tree.

Application of MST:

1) ~~Fram~~ telephone writing system through diffent ~~dooing~~
2) Electrical network delighty
3) electric analysis
4) traveling salesman moslew
5) handwriting recognition
6) img exglnatation.

② ① Krushal's Algorithm.

$TC = O(|E| \log(E))$

$SC = O(|V|)$

② Dijlistra Algorithm

$TC = O(V^2)$ (mouse a digit ~~Ary~~) $SC = O(V^2)$

$TC = O(E + V\log V)$ (Fibonaci heap PQ)

$TC = O(E \log V)$ (binary heap as PQ)